2020-07-02 Ontology Interest Group Call -- GitHub

Thursday, July 2, 2020, 10 AM US Eastern Time

Connection Info

To join the online meeting:

- Go to: https://ufl.zoom.us/j/94205948889?pwd=WG5DT2dpT0JjYU9lVFpJY04yOFdEZz09
- One tap mobile:
    - US: +16699006833,,9358074182# or +19292056099,,9358074182#
- Or Telephone:
    - US: +1 669 900 6833 or +1 929 205 6099 or 877 853 5257
    - Meeting ID: 935 807 4182
- International numbers available: https://zoom.us/u/aeANHanzED

Attendees

Resources

Ontology Interest Group Google Folder https://drive.google.com/drive/u/0/folders/1RGBh4fDZdzpJdwyiUMO8OPWwkcmVYrI0

Google Doc for meeting notes: https://bit.ly/2YPPBwq

Attendees

- Mike Conlon
- *your name here*

Agenda

1. Github for Ontology work
    a. Where?
    b. How/process: issues/forks/pull requests
    c. Useful background/training/help guides
    d. Questions, comments
    e. What's next?

Notes

1. Github for Ontology work
    a. VIVO Processes: https://github.com/vivo-project/VIVO/wiki/Git---GitHub-Processes
    b. General idea: It's just a bunch of files
    c. We work in the open
    d. Folder structure will be needed (ontology group can develop). ODK was too complex.
    e. Where?
        i. ADO in mconlon17/ado on Github
        ii. AEON in TIBonto/aeon on Github
        iii. LANG in vivo-community/language-ontology on Github
        iv. VIVO.owl (single file) in vivo-project/vivo on GitHub
        v. Github vivo-ontology-lab (organization) unused
    f. How/process: issues/forks/branches/pull requests
        i. Fork to personal Github
        ii. Clone personal fork to local
        iii. Edit the local clone
        iv. The remote is a connection from the local clone to github. You typically have a connection to your fork. You can create a remote to the canonical source.
        v. A branch creates a snapshot and is a safe place to work. Code is isolated.
        vi. Changes can be in three states:
            1. Local only
            2. Git add -- "plan to commit; staging"
            3. Git commit -- creates a bundle of changes to the fork
            4. Git pull request from the fork to the remote
        vii. Principles
            1. Never work in master
            2. Always work in a branch

3. Clean work: Issue -> branch -> push to fork -> pull request to remote; One issue at a time.
    a. Github desktop makes this easier.
4. A pull might have 8 commits -- fixing typos, making changes request by reviewers, etc. A single commit (squash merge produces a single commit to the master branch) keeps the master branch commit history clean -- each commit is associated with a unit of work.

g. Useful background/training/help guides
    i. A comparison of branching and forking: https://support.atlassian.com/bitbucket-cloud/docs/branch-or-fork-your-repository/
h. Some suggestions
    i. If you are not using GitHub desktop, it might be helpful
    ii. Recommend to review in detail the VIVO repo wiki document: https://github.com/vivo-project/VIVO/wiki/Git---GitHub-Processes
    iii. We're going to need to practice